

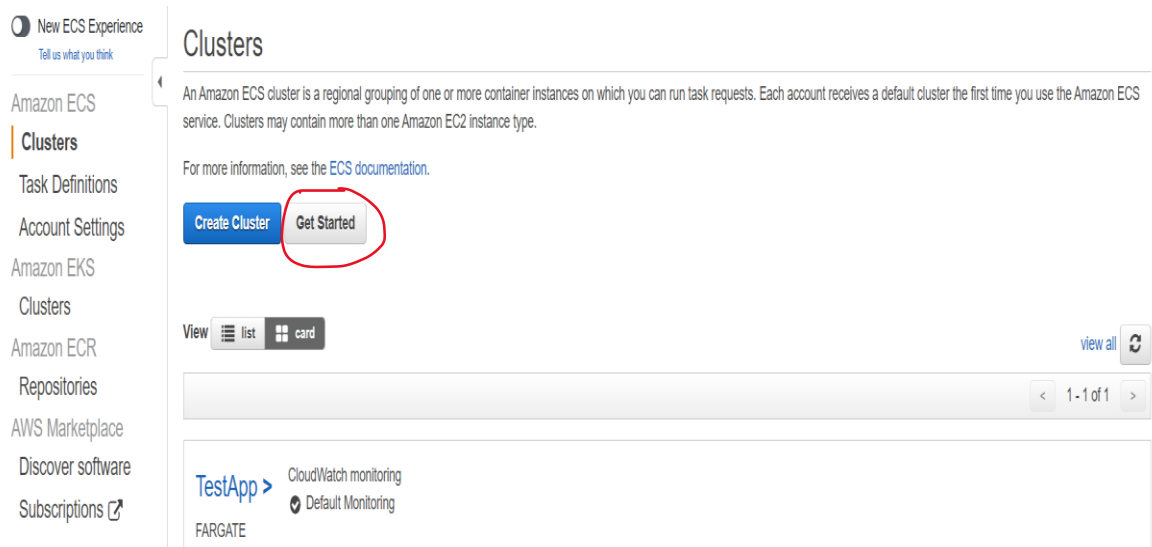
# HOSTING A DASHBOARD APPLICATION IN AWS FARGATE

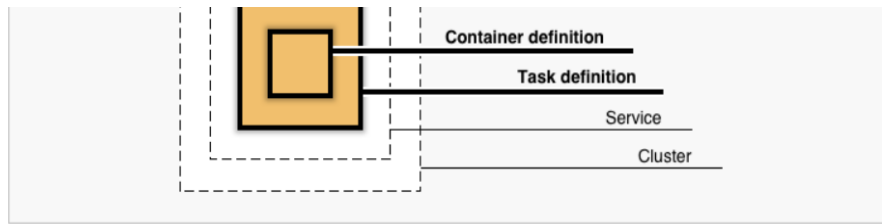
## PRE-REQUISITE :

- Create a docker image for dashboard application following the steps mentioned in the 'Docker-Dashboard' Readme.md file.
- Upload the docker image in AWS ECR.

## STEPS TO HOST THE APPLICATION IN AWS FARGATE

- **Deploy the Docker Container using ECS**  
The Elastic Container Service (ECS) runs and manages Docker containers. It is highly scalable, allowing more containers to be deployed automatically to meet demand. Return to the home page of the AWS management console and search for "ECS". Once in ECS, let's click on the Get Started button. You will be taken to Step 1: Container and Task. The container definition describes the requirements of your container as well as how the system should run your container. By clicking on the Configure button, we can configure the container to accommodate the dashboard app.





## Container definition

Edit

Choose an image for your container below to get started quickly or define the container image to use.

### sample-app

image : httpd:2.4  
memory : 0.5GB (512)  
cpu : 0.25 vCPU (256)

### nginx

image : nginx:latest  
memory : 0.5GB (512)  
cpu : 0.25 vCPU (256)

### tomcat-webserver

image : tomcat  
memory : 2GB (2048)  
cpu : 1 vCPU (1024)

### custom

image : --  
memory : --  
cpu : --

Configure

We only need to populate the first two fields. We name the container codesecure-app and we populate the second field with the Image URI from AWS ECR of the dashboard docker image. Click on Update.

features, marketplace

5 vCPU (256)

t-webserver

tomcat

2GB (2048)

1 vCPU (1024)

inition

on is a blueprint for the task level but

Task definition

Ne

Task ex

Co

T

## Edit container

▼ Standard

Container name\* codesecure-app ⓘ

Image\* public.ecr.aws/217-8-8-8/codesecure-app:latest ⓘ

Private repository authentication\* ☐ ⓘ

Memory Limits (MiB) Soft limit ▼ 128 ⓘ

➕ Add Hard limit

Define hard and/or soft memory limits in MiB for your container. Hard and soft limits correspond to the "memory" and "memoryReservation" parameters, respectively, in task definitions. ECS recommends 300-500 MiB as a starting point for web applications.

Port mappings

Container port	Protocol
<input type="text"/>	tcp ▼ ⓘ

\* Required

Cancel Update

Scroll down to the Task Definition and click Edit. Here, we can specify the hardware requirements for the container. We populate these fields as below and Save.

Configure task definition: codesecure-app

### Task definition details

Task definition name\* codesecure-app ⓘ

Network mode\* awsvpc ⓘ

Task execution role ecsTaskExecutionRole ⓘ

Compatibilities\* FARGATE ⓘ  
[Learn more about compatibilities](#)

### Task size

Task size allows you to size at the task level and optionally set container-specific CPU and memory sizes. You are billed for the task memory and task CPU allocated.

Task memory\* 1GB (1024) ▼

Task CPU\* 0.5 vCPU (512) ▼

\*Required

Cancel Save

When we click on Next, we are taken to the Service Definition page. These fields are pre-populated, so click on Next to go to the Cluster definition. All we need to do here is name the cluster codesecure-app and click Next. You can then review the configuration and click Create.

Container definition  
Task definition  
Service  
Cluster

### Configure your cluster

The infrastructure in a Fargate cluster is fully managed by AWS. Your containers run without you managing and configuring individual Amazon EC2 instances.

To see key differences between Fargate and standard ECS clusters, see the [Amazon ECS documentation](#).

Cluster name codesecure-app

Cluster names are unique per account per region. Up to 255 letters (uppercase and lowercase), numbers, and hyphens are allowed.

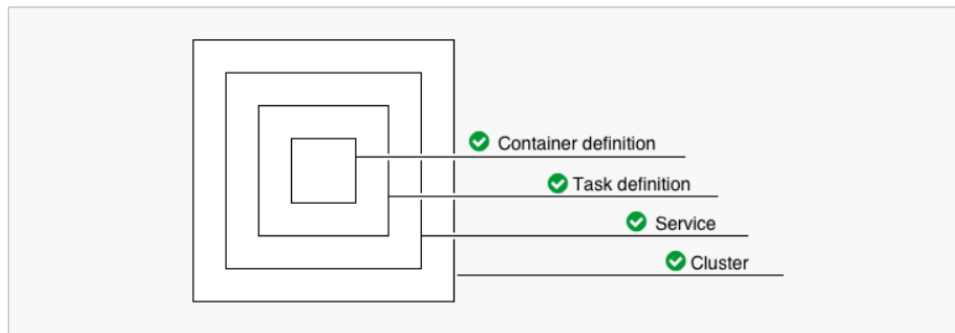
VPC ID Automatically create new ⓘ

Subnets Automatically create new ⓘ

\*Required

Cancel Previous Next

## Diagram of ECS objects and how they relate



## Review

Review the configuration you've set up before creating your task definition, service, and cluster.

### Task definition

[Edit](#)

Task definition name codesecure-app

Network mode awsvpc

Task execution role Create new

Container name codesecure-app

Image public.ecr.aws/~~com.amazonaws~~/codesecure-app:latest

Memory 1024

Port -

Protocol HTTP

### Service

[Edit](#)

Service name codesecure-app-service

Number of desired tasks 1

### Cluster

[Edit](#)

Cluster name codesecure-app

VPC ID Automatically create new

Subnets Automatically create new

\*Required

[Cancel](#)[Previous](#)[Create](#)

The process will take a good few minutes. Once completed, click View Service.

- Accessing the Dashboard Service

By default, the service will only allow traffic on port 80. This is a problem as the Docker container is only accessible on port 8050. We thus need to change the Security groups rules. Click on the Security groups identifier

Cluster `codesecure-app`
Status ACTIVE
Task definition `codesecure-app:8`
Service type `REPLICA`
Launch type `FARGATE`
Service role `AWSServiceRoleForECS`
Created By  
arn:aws:iam::`000000000000`:role/`codesecure-app`  
2/AWSReservedSSO\_CIC-AWSPowerUserAccess\_`000000000000`

Desired count `1`  
Pending count `1`  
Running count `0`

Details Tasks Events Auto Scaling Deployments Metrics Tags Logs

Load Balancing

Load Balancer Name	Container Name	Container Port
No load balancers		

Network Access

Allowed VPC `vpc-000000000000`
Allowed subnets `subnet-000000000000`, `subnet-000000000000`
Security groups\* `sg-000000000000`
Auto-assign public IP `ENABLED`

Once the page loads, click on Inbound Rules tab and the Edit inbound rules button.

Details

Security group name  
EC2ContainerService-codesecure-app-EcsSecurityGroup-000000000000

Security group ID  
sg-000000000000

Description  
ECS Allowed Ports

VPC ID  
vpc-000000000000

Owner  
000000000000

Inbound rules count  
1 Permission entry

Outbound rules count  
1 Permission entry

Inbound rules Outbound rules Tags

You can now check network connectivity with Reachability Analyzer
Run Reachability Analyzer

Inbound rules (1/1)

Filter security group rules

Manage tags Edit inbound rules

	Name	Security group rule...	IP version	Type	Protocol	Port range
<input checked="" type="checkbox"/>	-	sg-000000000000	IPv4	HTTP	TCP	80

Re-populate the fields as follows before clicking on Save rules.

### Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

**Inbound rules** Info

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-██████████	Custom TCP	TCP	8050	Custom	

0.0.0.0/0

Add rule

Cancel Preview changes Save rules

We can check if the service is running by clicking on Clusters in the ECS sidebar and clicking on codesecure-app . Once the page loads, click on the Tasks tab and subsequently click on the task identifier.

Details Tasks Events Auto Scaling Deployments Metrics Tags Logs

Last updated on October 21, 2021 5:58:36 AM (0m ago)

Task status: Running Stopped

Filter in this page

1-1 Page size 50

Task	Task Definition	Last status	Desired status	Group	Launch type	Platform version
37ec3cadf60443d6bc8d6c1265902b13	codesecure-app:8	RUNNING	RUNNING	service:codesecure-ap...	FARGATE	1.4.0

The task configurations will be shown.

Task : 37ec3cadf60443d6bc8d6c1265902b13
Run more like this Stop

Details Tags Logs

Cluster codesecure-app
Launch type FARGATE
Platform version 1.4.0
Task definition codesecure-app:8
Group service:codesecure-app-service
Task role None
Last status RUNNING
Desired status RUNNING
Created at 2021-10-21 05:54:44 +1100
Started at 2021-10-21 05:56:09 +1100

Network
Network mode awsvpc
ENI Id eni-██████████
Subnet Id subnet-██████████
Private IP 10.0.1.204
Public IP 18.142.48.238
Mac address 02:11:1d:85:01:90

Containers
Last updated on October 21, 2021 5:59:06 AM (0m ago)

Name	Container Runtime I...	Status	Image	Image Digest	CPU Units	Hard/Soft memory ...	Essential	Resource ID
codesecur...	██████████	RUNNING	public.ecr.aws/██████████esecu...		0	--	true	██████████

We can see that the task is running and is accessible at the address <http://18.142.48.238:8050> . Of course, your IP will be different. Substitute your IP in place of the above IP and you should be able to access your dashboard.